Course: Big Data

Lab 04

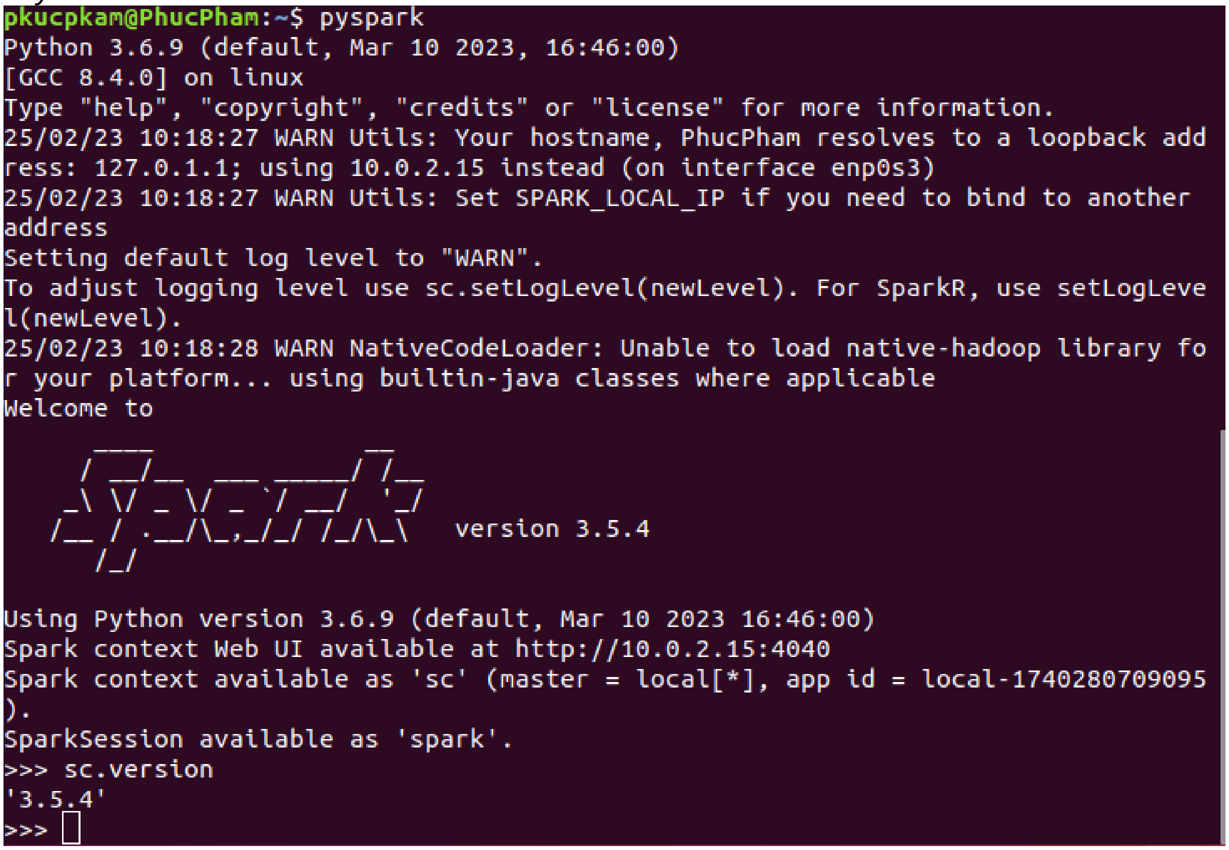
PySpark - RDD

Question 1:

Based on the tutorial of PySpark, students install PySpark in Ubuntu.

- Define the environment variable: JAVA_HOME
- Define the environment variable: SPARK_HOME
- Start the pyspark-shell and write an instruction to print down the PySpark version
- Take the screenshot and insert it into the table below.

My screenshot



```
pkucpkam@PhucPham:~$ pyspark
Python 3.6.9 (default, Mar 10 2023, 16:46:00)
[GCC 8.4.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
25/02/23 10:18:27 WARN Utils: Your hostname, PhucPham resolves to a loopback add
ress: 127.0.1.1; using 10.0.2.15 instead (on interface enp0s3)
25/02/23 10:18:27 WARN Utils: Set SPARK_LOCAL_IP if you need to bind to another
address
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLeve
l(newLevel).
25/02/23 10:18:28 WARN NativeCodeLoader: Unable to load native-hadoop library fo
r your platform... using builtin-java classes where applicable
Welcome to
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /__ / .__/\_,_/_/ /_/\_\   version 3.5.4
      /_/

Using Python version 3.6.9 (default, Mar 10 2023 16:46:00)
Spark context Web UI available at http://10.0.2.15:4040
Spark context available as 'sc' (master = local[*], app id = local-1740280709095
).
SparkSession available as 'spark'.
>>> sc.version
'3.5.4'
>>> 
```

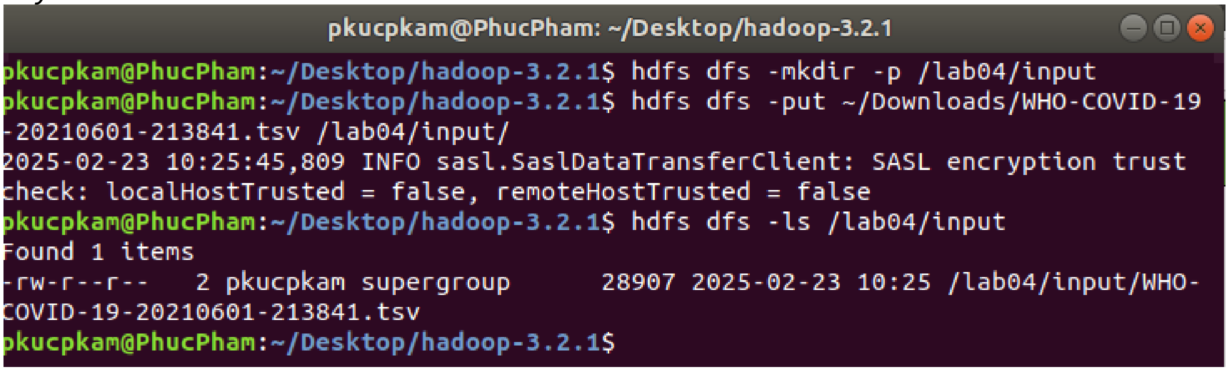
Question 2:

Given a tsv file WHO-COVID-19-20210601-213841.tsv which is corresponding to the WHO Coronavirus (COVID-19) Dashboard.

Students are required to create a folder, named **lab04**, in HDFS and then copy the tsv to **lab04/input/**

Take a screenshot to show the content of **lab04/input/** in HDFS

My screenshot



pkucpkam@PhucPham: ~/Desktop/hadoop-3.2.1

```
pkucpkam@PhucPham:~/Desktop/hadoop-3.2.1$ hdfs dfs -mkdir -p /lab04/input
pkucpkam@PhucPham:~/Desktop/hadoop-3.2.1$ hdfs dfs -put ~/Downloads/WHO-COVID-19
-20210601-213841.tsv /lab04/input/
2025-02-23 10:25:45,809 INFO sasl.SaslDataTransferClient: SASL encryption trust
check: localHostTrusted = false, remoteHostTrusted = false
pkucpkam@PhucPham:~/Desktop/hadoop-3.2.1$ hdfs dfs -ls /lab04/input
Found 1 items
-rw-r--r--   2 pkucpkam supergroup      28907 2025-02-23 10:25 /lab04/input/WHO-
COVID-19-20210601-213841.tsv
pkucpkam@PhucPham:~/Desktop/hadoop-3.2.1$
```

Question 3:

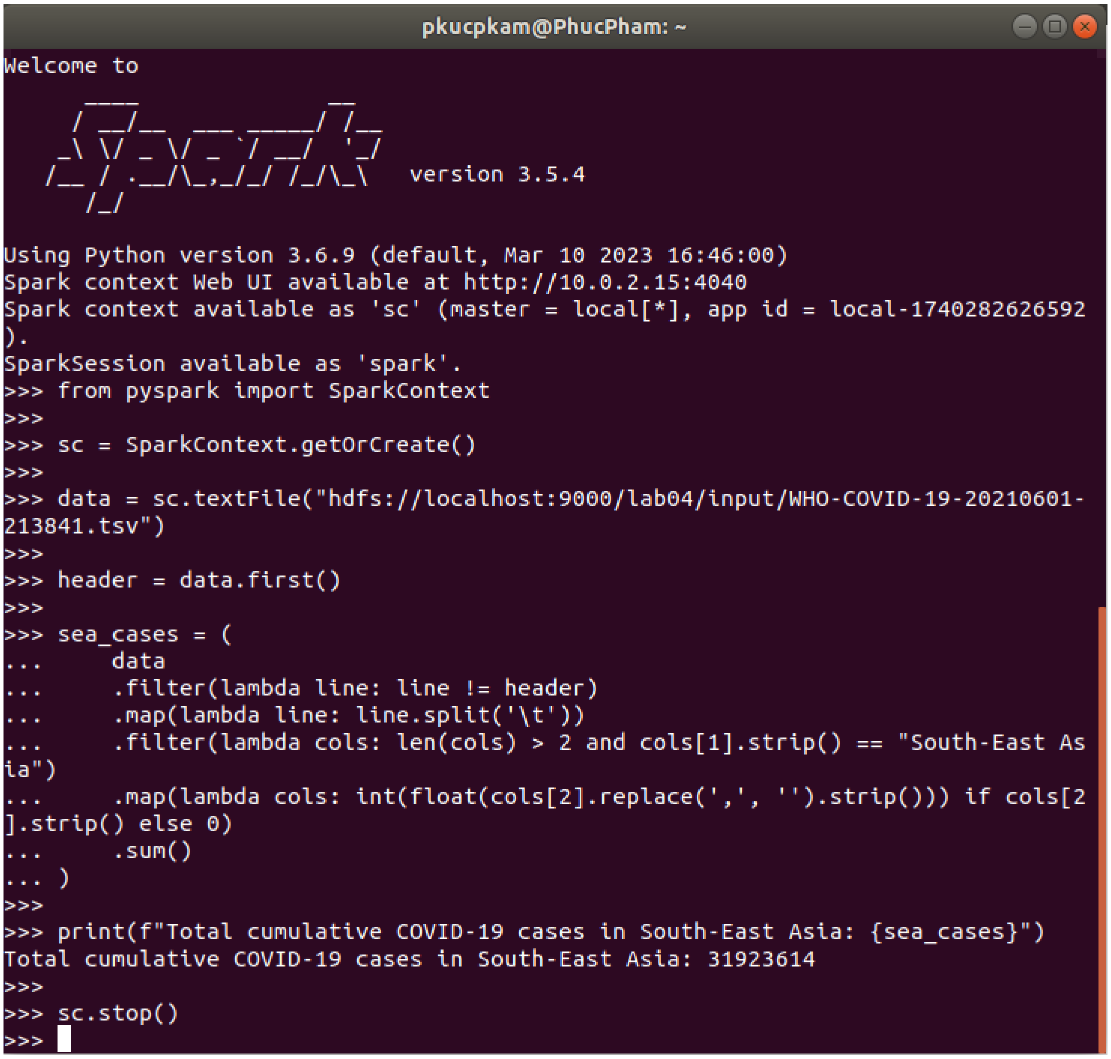
Write a PySpark program, located in **ASEANCaseCount.py**, to count the number of cumulative total cases among ASEAN countries (*South-East Asia Region in the given data table*) using RDDs.

- Insert your source code into the table below.

```
from pyspark import SparkContext

sc = SparkContext.getOrCreate()

data = sc.textFile("hdfs://localhost:9000/lab04/input/WHO-COVID-19-
20210601-213841.tsv")

header = data.first()

sea_cases = (
    data
    .filter(lambda line: line != header)
    .map(lambda line: line.split('\t'))
    .filter(lambda cols: len(cols) > 2 and cols[1].strip() == "South-
East Asia")
    .map(lambda cols: int(float(cols[2].replace(',', '').strip())) if
cols[2].strip() else 0)
    .sum()
)

print(f"Total cumulative COVID-19 cases in South-East Asia:
{sea_cases}")

sc.stop()
```

- Take a screenshot of the terminal to visualize the program result.

My screenshot



```
Welcome to
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /__ / .__/\_,_/_/ /_/\_\   version 3.5.4
      /_/

Using Python version 3.6.9 (default, Mar 10 2023 16:46:00)
Spark context Web UI available at http://10.0.2.15:4040
Spark context available as 'sc' (master = local[*], app id = local-1740282626592
).
SparkSession available as 'spark'.
>>> from pyspark import SparkContext
>>>
>>> sc = SparkContext.getOrCreate()
>>>
>>> data = sc.textFile("hdfs://localhost:9000/lab04/input/WHO-COVID-19-20210601-
213841.tsv")
>>>
>>> header = data.first()
>>>
>>> sea_cases = (
...     data
...     .filter(lambda line: line != header)
...     .map(lambda line: line.split('\t'))
...     .filter(lambda cols: len(cols) > 2 and cols[1].strip() == "South-East As
ia")
...     .map(lambda cols: int(float(cols[2].replace(',', '').strip())) if cols[2
].strip() else 0)
...     .sum()
... )
>>>
>>> print(f"Total cumulative COVID-19 cases in South-East Asia: {sea_cases}")
Total cumulative COVID-19 cases in South-East Asia: 31923614
>>>
>>> sc.stop()
>>>
```